Received: December 19, 2019. Revised: February 12, 2020.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Economic Order Picker Routing Considering Travel Time and Vehicular Energy Consumption with Varying Aisle Traffic

Theerawat Rojanapitoon[1]* **Jirarat Teeravaraprug**[1]

Faculty of Engineering, Thammasat University, Pathum Thani, 12121, Thailand
* Corresponding author's Email: dieobox@gmail.com

Abstract: Warehousing can be expensive because order picking requires considerable vehicular movement and labor hours. Although many previous studies have focused on order picker routing, there exists a lack of research on the simultaneous increase of order picking speed and energy reduction in rectangular warehouses with varying levels of traffic in each aisle. This study accordingly developed and evaluated a mathematical model for determining optimal picker routes considering the total travel time and energy consumed. The results were validated using the brute-force search method and benchmarked with the time-staged (TS) model. The energy savings were determined by comparing a time-optimized use-case (T) with one optimized for both time and energy (TE). Both use-cases provided routes up to 44% faster than the TS and avoided more than 50% of congested paths, and TE which represents the full functionality of our model provided a possible energy savings of up to 17% over T.

Keywords: Energy, Mathematical model, Order picking, Routing, Warehouse.

1. Introduction

Warehouses play a critical role in the supply chain, enabling the efficient movement of goods. Warehouse operations include receiving, transferring and stocking, order picking, accumulation and sortation, and cross-docking and shipping. Order picking has long been identified as the most labor-intensive, time-consuming and costly activity in practically every warehouse, consuming up to 60% of all labor hours [1], and as much as 55% of the total warehouse operating costs [2]. Moreover, pickers spend 60% of the order-picking time on traveling [1]. Therefore, the order picking process is given the highest priority by warehousing professionals to improve productivity. Besides productivity improvements, nowadays, either due to government regulation or market demand, most organizations are beginning to go green. Energy-saving efforts are also identified as an important task for green supply chain implementation [3-4]. For these reasons, businesses must improve order picking speed and reduce energy use simultaneously.

During the order-picking process, pickers often travel with their powered vehicles (fuel-type or electric) to gather specified amounts of the appropriate stock-keeping units (SKUs) according to an order. Although this may seem like an easy task, in large warehouses, pickers may be unaware of strategies to complete their trips efficiently. Especially when having to consider several factors like travel time, energy consumption, and traffic conditions. Thus, appropriate routing could not only increase the efficiency of order picking but also reduces the energy consumption of vehicles. Additionally, a small energy saving in a single vehicle would imply a large energy saving for the entire fleet [5].

Although many previous studies have focused on order picker routing, there exists a lack of research on the simultaneous increase of order picking speed and energy reduction in rectangular warehouses with varying levels of traffic in each aisle. This finding motivated our study.

This study attempts to determine an optimal order picker route by considering both travel time and vehicular energy consumption. In this scheme, the pickers travel with their vehicles in a rectangular warehouse in which the travel speed in each aisle of the warehouse may vary due to aisle congestion, and only a single picker is considered for each round of computation. A multi-objective mathematical model to determine an optimal route with consideration of travel time and energy consumption was developed that has the ability to simultaneously increase the order picking speed (thus providing travel time savings) and reduce energy consumption in rectangular warehouses with varying levels of traffic in each aisle. Further, a warehouse traffic map that provides historical travel speeds in each aisle to interoperate with the model was utilized. A computer simulation software was then applied to solve sample problems. The optimality of the results determined using the new model was then validated by the brute-force search method, which selects the optimal solution by evaluating all possible solutions, as well as comparison with the results provided by another routing model used as a baseline. Additionally, a numerical example is provided to verify the performance of the model and illustrate its utility.

The remainder of this paper is organized as follows: Section 2 reviews some related research and presents our research motivations, Section 3 describes the characteristics of the example warehouse and problem scope, Section 4 presents the proposed novel mathematical model and the experimental design used to validate its optimality and performance, Section 5 provides the results and a discussion, and the conclusions and future directions of this work are presented in Section 6.

2. Literature review

Previous research into order picker routing has focused on either heuristic or optimal strategies for manual or automated warehouses. Heuristic strategies such as the S-shape, largest gap, return, and composite heuristics are widely used in practice as they are easy to implement [1], Many recent studies apply these strategies with the modern complicated problem such as the routing with access restriction problem [6], the routing in a non-standard warehouse [7] or the considering of both storage assignments and routing [8]. Although the efficient solution could be obtained, they are not guaranteed to return an optimal solution [1, 9]. Moreover, the performance of a heuristic routing strategy depends on the number of picks per aisle, it is impossible to establish a policy that achieves superior performance in all scenarios [10].

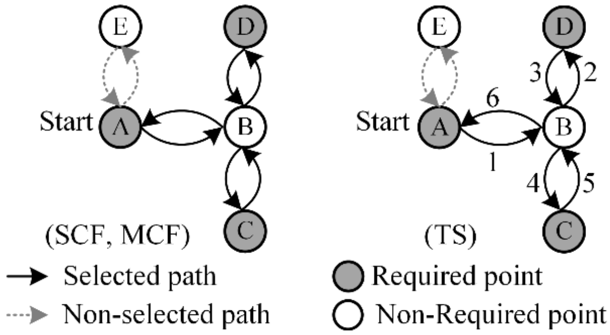


Figure.1 The solutions of the SCF and MCF formulations (left) compared with the solution of the TS formulation (right)

In contrast to the heuristic routing strategies, optimal routing guarantees an optimal solution for each tour [1, 9]. These types of problems are typically formalized as the Steiner traveling salesman problem (STSP), a variant of the traveling salesman problem (TSP). Numerous studies have focused on optimal routing methods, several of which emphasize the use of either distance or travel time as the criterion for defining optimal routing [11-17]. Roodbergen and De Koster [11] proposed a dynamic programming method to determine the shortest order picking routes in three cross-aisles (two blocks) warehouse. This approach is restricted to implementation in more complicated order picking problems [11, 12].

However, it is possible to determine the optimal routes for both basic and more complicated order picking problems by solving the instance of the integer programming model [12, 13]. Letchford, Nasiri, and Theis [13] proposed three compact integer programming formulations for the STSP problem: the single commodity flow (SCF), multi commodity flow (MCF), and time-staged (TS). Among the three compact formulations, the calculation performances of SCF and MCF are better but, as shown in Fig. 1, the TS is the only formulation that provides the exact sequence of traveling paths; other formulations only show which paths should be selected.

Recently, many studies have focused on realistic picking contexts by simultaneously considering many criteria [14-17]. To address such multi-criteria problems, most researchers have focused on using multi-objective modeling to formulate the objective function and constraints, then solving by exact or approximation algorithms.

No previous studies have simultaneously considered the travel time and energy consumption of the picker vehicle in a warehouse, in which the picker must travel through aisles with varying levels of traffic. The advantages of the integer programming approach and multi-objective modeling motivated us to construct a mathematical model formulated as a multi-objective integer programming model that interoperates with a newly introduced warehouse traffic map to determine optimal routing.

The proposed model was applied to two use-case scenarios: one considering only time (T) and the other considering both time and energy (TE), the comparison between which was used to verify the energy savings performance of the fully functional TE use-case. We also compared the results of the proposed model with the TS formulation proposed by Letchford, Nasiri, and Theis [13], which uses only the travel distance as its optimization criterion. This formulation could be used as the baseline for comparison as it is a single STSP formulation that gives the exact sequence of the optimal path, which is essential for determining the necessary travel steps.

Moreover, to confirm that the solution obtained by solving the models proposed in this study is truly optimal and unaffected by any mistake in the solver package or programming code, several optimization studies employing the brute-force search method [18-21] were used as a baseline. The brute-force search method enumerates all possible candidate solutions and then checks whether each candidate satisfies the problem statement. This type of method is ideal for confirming optimality because it always obtains a solution (if there is one) and is simple to implement without a sophisticated formulation.

3. Problem description

This paper considers a low-level picker-to-parts order-picking in a rectangular warehouse with parallel and cross aisles, as illustrated in Fig. 2. The area between two adjacent cross aisles is called a block, and each block has 4 double-sided shelves, each shelf has 12 storage locations (six on each side). Pickers can pass each other in opposite directions in an aisle. The point from which the picker starts a trip and returns to deposit picked items, called the depot, is located at the bottom-left corner of the warehouse. Gray areas show where the required items are placed.

Fig. 3 shows a graph representation of the problem. The nodes are represented by numbers and possible routes by arrows. Gray nodes denote the location of required items, and node 0 denotes the depot. Although there may be multiple pickers in the warehouse at any given time, this paper considers only one picker at a time. The picker uses an electric order picking vehicle to travel, as illustrated by Fig. 4, which has a navigation system that provides the picker with important information, including the routing plan and the picklist.

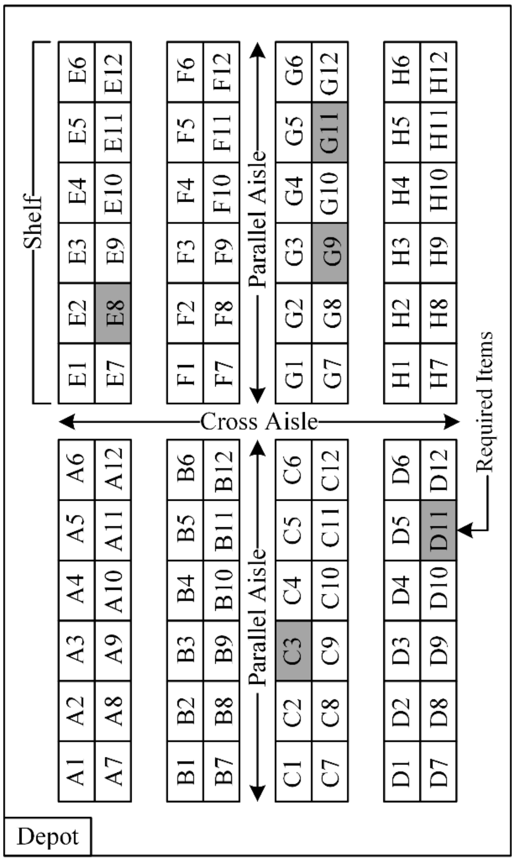


Figure.2 The sample warehouse

The picklist is a list of required items, which may have different weights. After the required items are picked, the picker and vehicle return to the depot to deposit all picked items and stand by for a new trip. Some aisles in the warehouse may be crowded, requiring the picker to travel through at a limited speed. The traffic map shown in Fig. 5 is used to provide information relating to traffic conditions in the warehouse: clear traffic routes are indicated by solid lines (S1), and congested routes are indicated by dashed lines (S2). This traffic map is assumed for our simulation, but in real-life applications, a traffic map can be formulated using the average speed of pickers that previously traverse the aisle. These data could be collected from an indoor positioning system attached to vehicles. The total energy consumption considers only energy-related to routing decisions, rather than real total energy. We assume that vehicles immediately have a certain velocity from start moving, the acceleration at starting and stopping is not considered due to it depends on the driving of pickers.

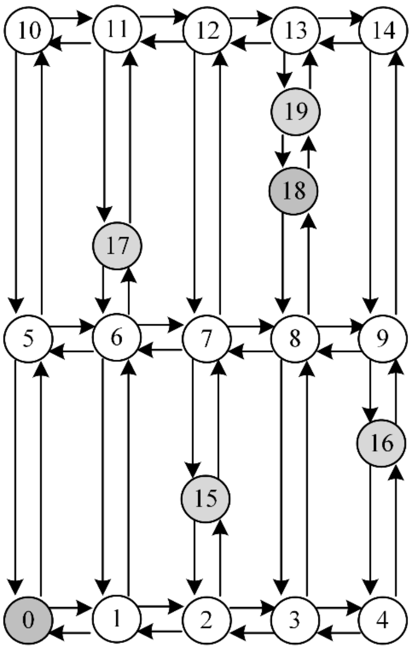


Figure.3 Graph representation of the sample warehouse

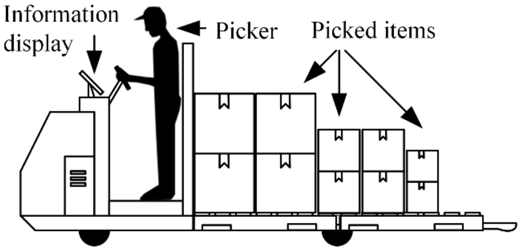


Figure.4 Order picking vehicle

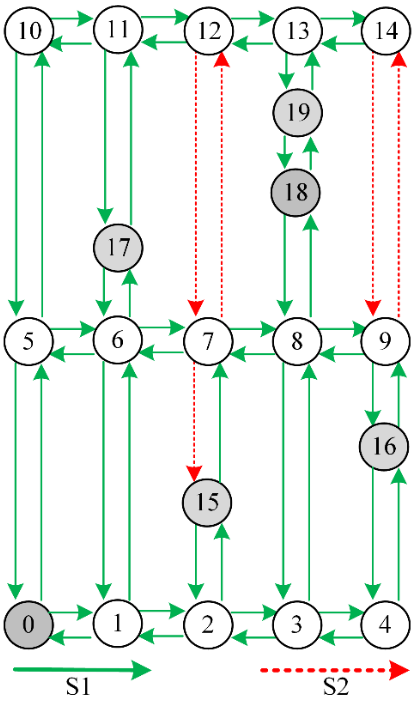


Figure.5 Traffic map of the example warehouse

Table 1. Notations

Variable	Meaning
d_a	Travel distance of arc a (m)
v_a	Average speed of previous vehicles that passed arc a (m/s)
r_a^k	binary variable = 1 when arc a is selected to travel in step k = 0 otherwise
f_a^k	Transporting weights passing by arc a in step k (N)
W_c	Weight of picker and vehicle (N)
W_T	Weight of all items on the pick list (N)
w_i	Weight of item to be picked at vertex i (N)
w	Objective weight
$\delta^+(i)$	Outgoing arcs of vertex i
$\delta^-(i)$	Incoming arcs of vertex i
k	Step index
A	Set of arcs
V	Set of all vertices, including the depot, intersections, corners, and required item locations
V_R	Set of required vertices (depot and required item locations)
μ_r	Rolling resistance coefficient
F	Force to drive vehicle (N)
m_T	Total mass of vehicles, including mass of vehicle, picker and carrying SKUs (kg)
acc	Acceleration of vehicle (m/s^2)
g	Acceleration due to gravity (m/s^2)

4. Mathematical model

All notations are listed in Table 1. Based on the graph representation, as seen in Fig. 3, the newly formulated model, combines two functions: total travel time and total energy consumption function. The total travel time function, expressed in Eq. (1), it is the sum of the products of the selected arc r_a^k and its distance d_a divided by the historical average speed v_a of arc a. Step k increases by 1 when the picker moves from the current vertex to another vertex and is limited to 2(|V|-1) because there exists an optimal STSP solution for which the total number of steps does not exceed 2(|V|-1), this theorem was proved by Letchford, Nasiri, and Theis [13].

$$Total\ Travel\ time = \sum_{k=1}^{2(|V|-1)} \sum_{a\in A} r_a^k \frac{d_a}{v_a} \tag{1}$$

The total energy consumption function expressed in Eq. (2) considers only energy-related routing decisions, rather than total energy use.

$$Total\ Energy\ = \sum_{k=1}^{2(|V|-1)} \sum_{a\in A} \mu_r (r_a^k W_c +\ f_a^k) d_a \tag{2}$$

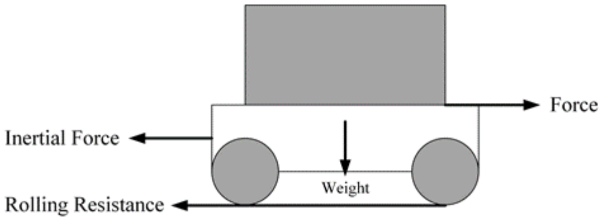


Figure.6 The motion resistance of the vehicle

Further, for this function, which is based on Newton's laws, the vehicle energy consumption equivalent to the minimum work done by the vehicle to carry the load along the path, is shown in Eq. (3).

$$Energy = Work = F.d_a \tag{3}$$

Since the vehicle uses wheels for motion, the minimum force necessary to drive the vehicle is related by the inertial force and rolling resistance, as seen in Fig. 6 and can be express by Eq. (4).

$$F = m_T.acc + \mu_r.m_T.g \tag{4}$$

According to the study's assumption, the inertial forces ($m_T.acc$) term are not considered, Eq. (4) thus becomes Eq. (5)

$$F = \mu_r.m_T.g \tag{5}$$

Substitutes Eq. (5) into Eq. (3) yields:

$$Energy = \mu_r.m_T.g.d_a \tag{6}$$

For Integer programming formulation, the weight ($m_T.g$) in Eq. (10) is replaced by $r_a^k W_c + f_a^k$, where $r_a^k W_c$ is the weight of vehicle passing through arc a in step k, if arc a was selected, while f_a^k is a linking variable (carried weight) of r_a^k, thus Eq. (6) becomes Eq. (7).

$$Energy = \mu_r(r_a^k W_c + f_a^k)d_a \tag{7}$$

Therefore, the total energy consumption function can be formulated as seen in Eq. (2). Eq. (1) and (2) are combined and normalized to form a new objective function, shown in Eq. (8). The model's constraints are expressed in Eq. (9) – (16). Constraints (9) and (10) ensure that the picker first departs from the depot and that no other arcs are selected in the first step. Constraint (11) ensures that all required vertices are visited at least once. Constraint (12) states that the picker arrives at and then departs from all vertices (except the depot) in sequential steps. Constraint (13) is a load conservation constraint. Constraint (14) ensures that vehicle load will be increased when the picker retrieves an item. Constraint (15) ensures that if

$$Z = min \quad w\,\frac{\sum_{k=1}^{2(|V|-1)}\sum_{a\in A} r_a^k \frac{d_a}{v_a}}{\sum_{a\in A}\frac{d_a}{v_a}} + (1-w)\frac{\sum_{k=1}^{2(|V|-1)}\sum_{a\in A} d_a\mu_r(r_a^k W_c + f_a^k)}{\sum_{a\in A} d_a\,\mu_r(W_c + W_T)} \tag{8}$$

$$s.t. \sum_{a\in\delta^+(0)} r_a^1 = 1 \tag{9}$$

$$\sum_{a\in A} r_a^1 = 0 \qquad (\, a \in A \setminus \delta^+(0)\,) \tag{10}$$

$$\sum_{k=1}^{2(|V|-1)} \sum_{a\in\delta^+(i)} r_a^k = 1 \qquad (\,\forall\, i\, \epsilon\, V_R\,) \tag{11}$$

$$\sum_{a\in\delta^-(i)} r_a^k = \sum_{a\epsilon\delta^+(i)} r_a^{k+1} \qquad (\,\forall\, i\, \epsilon\, V \setminus \{0\}; 1 \le\ k\ \le 2(|V|-1)-1) \tag{12}$$

$$\sum_{a\in\delta^+(i)} f_a^{k+1} - \sum_{a\epsilon\delta^-(i)} f_a^k = 0 \qquad (\,\forall\, i\, \epsilon\, V \setminus V_R;\ 1 \le\ k\ \le 2(|V|-1)-1) \tag{13}$$

$$\sum_{a\in\delta^+(i)} f_a^{k+1} - \sum_{a\epsilon\delta^-(i)} f_a^k\ = w_i \sum_{a\epsilon\delta^+(i)} r_a^k \qquad (\,\forall\, i\, \epsilon\, V_R \setminus \{0\};\ 1 \le\ k\ \le 2(|V|-1)-1) \tag{14}$$

$$0 \le\ \ f_a^k\ \le\ W_T r_a^k \qquad (\,\forall\, a\, \epsilon\, A \setminus \delta^+(0);\ 1 \le\ k\ \le 2(|V|-1)) \tag{15}$$

$$r_a^k \in \{0{,}1\} \qquad (\,\forall\, a \in A\,;\ 1 \le\ k\ \le 2(|V|-1)\,) \tag{16}$$

transporting weights pass by arc a, then that arc is selected. Constraint (16) ensures that all decision variables are binary. Subtours cannot occur, since only one arc from the depot can be selected (constraints (9) and (10)), and arcs are continuously connected to another selected arc in subsequent steps (constraint (12)) until every required vertex V_R has been visited (constraint (11)).

When considering only time, we set the objective weight w to 1, which causes the energy consumption term to be zero. When considering both time and energy, total travel time minimization should be set as a priority while simultaneously using the least amount of energy. Hence, the objective weight w is set to 0.9 for this use-case.

The TS formulation proposed by Letchford, Nasiri, and Theis [13] was used as the baseline for evaluation of the model proposed in this study. The objective function and constraints were defined as shown in Eqs. (17) – (22), in which Eq. (17) is the objective function and Eqs. (18) – (22) are its constraints: Eqs. (18) – (19) ensure that the picker starts at the depot and no arc is selected in the first step, Eq. (20) ensures that all required vertices (including the depot) are visited, Eq. (21) ensures that the picker enters and leaves from any vertex in the next step and Eq. (22) ensures that all decision variables are binary.

5. Simulations

After formulating the model, we developed software to simulate and display the results. The software was built with C# on the .NET 4.0 framework in combination with Python 2.7 and the Pymprog 1.0 simplex solver library, using the Visual Studio 2015 Community Edition as an integrated development environment. The computer used for this experiment consisted of an Intel Core i7 CPU with 16GB memory. The software process diagram is shown in Fig.7.

From the outset, the user has a choice to generate a new order picking problem or load a previously saved problem. The generated problem includes the list of required items, picking locations, the weight of each item, and the generated warehouse traffic condition (average speed for each path). The user also has a choice to select the mathematical models between our proposed model and the Time-staged (TS) model. After finishing the step above, the software performs load calculation parameters that have been previously configured and then generates the traveling cost matrix formulated from the configuration of the warehouse that includes the number of shelves, the number of storage locations, aisle length, aisle spacing, and the traffic condition. The objective and its constraints are automatically loaded into the solver, and the solver is run to find the optimal route. On completion, the optimal route with calculated total travel time and the total energy consumption is displayed. The user is then presented with a choice to cross-check the simulation's result by running a Brute-force search method.

$$\boldsymbol{Z} = \boldsymbol{min} \sum_{k=1}^{2(|V|-1)} \sum_{a \in A} r_a^k \, d_a \tag{17}$$

$$\boldsymbol{s.t.} \sum_{a \in \delta^+(0)} r_a^1 = 1 \tag{18}$$

$$\sum_{a \in A} r_a^1 = 0 \qquad (\, a \in A \setminus \delta^+(0)\,) \tag{19}$$

$$\sum_{k=1}^{2(|V|-1)} \sum_{a \in \delta^+(i)} r_a^k = 1 \qquad (\, \forall\, i \in V_R \,) \tag{20}$$

$$\sum_{a \in \delta^-(i)} r_a^k = \sum_{a \epsilon \delta^+(i)} r_a^{k+1} \qquad (\forall\, i \in V \setminus \{0\}; 1 \leq\, k \,\leq 2(|V|-1)-1) \tag{21}$$

$$r_a^k \in \{0,1\} \qquad (\, \forall\, a \in A\,;\;\; 1 \leq\, k \,\leq 2(|V|-1)\,) \tag{22}$$

Received: December 19, 2019. Revised: February 12, 2020.

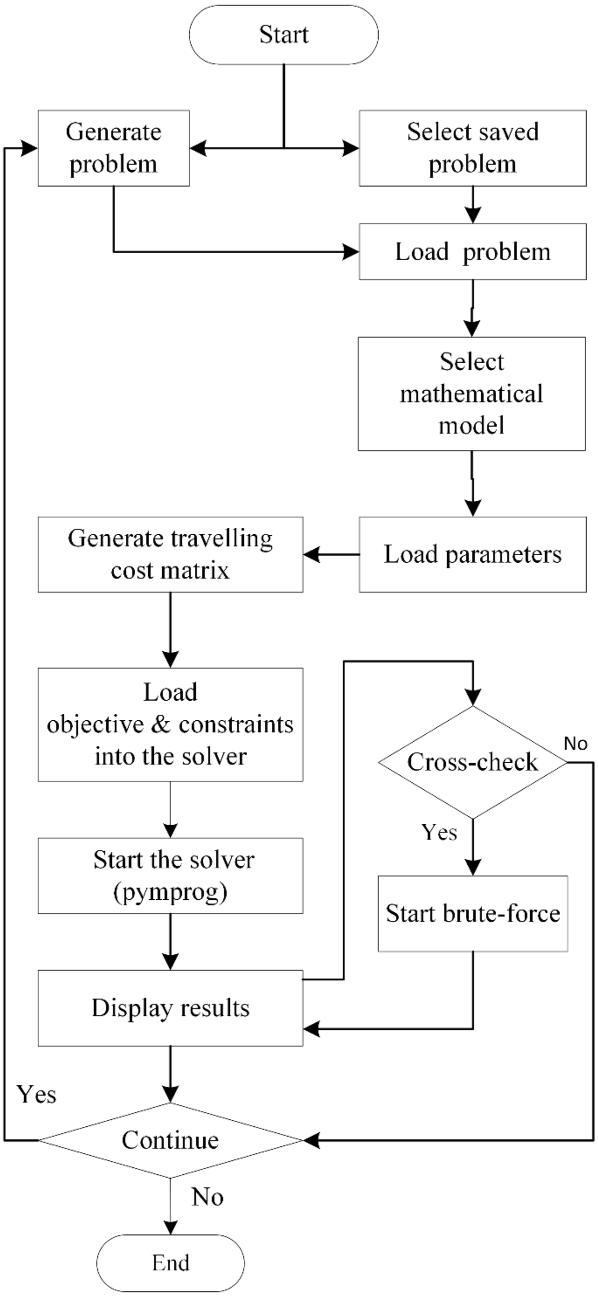


Figure.7 The software process diagram

Normally, this check returns the same result as that of the simulation, if the result is truly optimal, but may return a better solution if one exists. After the result is returned, the user has the option to continue with the next problem or end the operation.

Concerning the brute-force search method, ordinarily, this technique consumes a lot of time and resources. However, in this cross-checking scenario, the number of candidate solutions and iteration steps can be reduced since the total travel time, and the total energy consumption is already known from the previous simulation and can be used to limit the search space, resulting in faster checking. A schematic of this algorithm is shown in Fig. 8.

Based on the algorithm presented in Fig. 8, the traveling step (K) is initially set to 1. A set of tours is populated, for example, at K = 1, all populated tours in the list have one travel step, e.g. {0->1},{0->5}, destination vertices (1,5) are both neighbors of the start vertex (0), at K = 2. All tours from the previous step are continuously connected, and if new branches are found, they also added to the list.

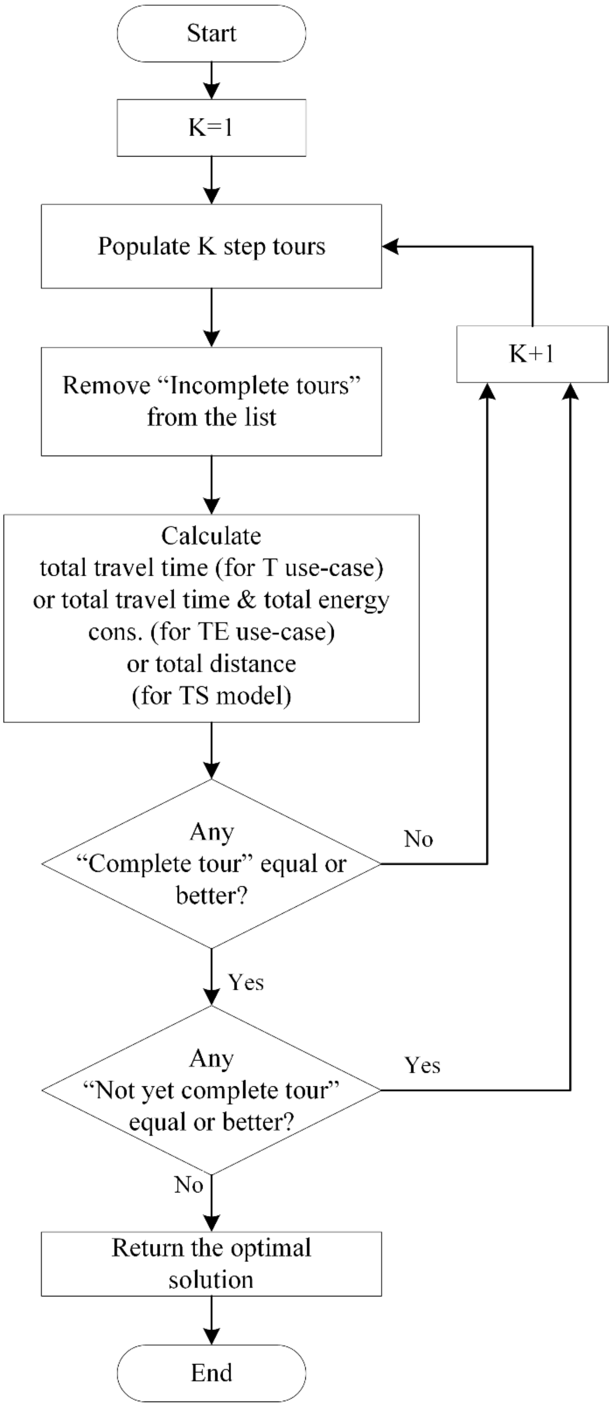
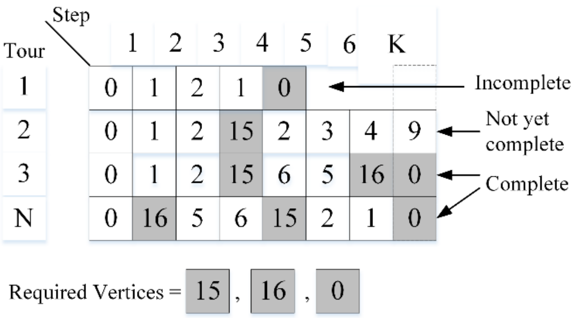

Start
K=1
Populate K step tours
K+1
Remove "Incomplete tours" from the list
Calculate total travel time (for T use-case) or total travel time & total energy cons. (for TE use-case) or total distance (for TS model)
Any "Complete tour" equal or better?
No
Yes
Any "Not yet complete tour" equal or better?
Yes
No
Return the optimal solution
End

Figure.8 Schematic representation of brute-force search method

Received: December 19, 2019. Revised: February 12, 2020.



Tour \ Step		1	2	3	4	5	6	K
1	0	1	2	1	0			
2	0	1	2	15	2	3	4	9
3	0	1	2	15	6	5	16	0
N	0	16	5	6	15	2	1	0

Figure.9 Characteristic of populated tours

Populated tours can be classified into three types as shown in Fig.9: "Incomplete tour." is the tour that finally returns to the depot (vertex 0) without visiting all the required vertices; "Not yet complete tour" is the tour that is still being completed and the "complete tour" is the tour that has already visited all the required vertices (including returning to the depot).

After each population process, all incomplete tours are removed from the list, and the total travel time and total energy consumption for each tour are calculated. Further, if there is no "Complete tour," the tour population process is continued. Otherwise, if any "Complete tour" has travel time and energy consumption equal or better than the simulation, the process continues checking whether any "Not yet complete tour" that has total travel time and total energy consumption equal or better than the previous optimal solution.

For the numerical experiments, all calculation parameters are shown in Table 2, and we set up 40 example problems with a different batch size; each item locations, congested paths, and weights are randomly assigned. These details are presented in Table 3.

Table 2. Simulation parameters

Parameter	Value
The objective weight w	0.9(TE),1(T)
Acceleration due to gravity g	9.8 m/s^2
Rolling resistance coefficient μ_r	0.1
Weight of vehicle and picker W_c	300 kg
Parallel aisle spacing	8 m
Parallel aisle depth	68 m
Vehicle's payload capacity	1,000 kg
Average speed of a clear (S1) route	1 m/s
Average speed of a congested (S2) route	0.1 m/s

We applied the proposed model to the T and TE use-cases in order to determine its energy saving performance. Then, the optimality of the calculation results was validated by comparison with the results obtained by the brute-force search method. After calculating the results of the T and TE use-cases using the proposed model, we then solved the same problems using the TS model to provide a baseline for comparison. Finally, the travel time savings percentage, energy savings percentage, and the congestion avoidance percentage were calculated using Eqs. (23) – (25).

$$Travel\ time\ savings\ \% = \frac{Total\ travel\ time_{TS} - Total\ travel\ time_{T\ or\ TE}}{Total\ travel\ time_{TS}} \times 100 \tag{23}$$

$$Energy\ savings\ \% = \frac{Energy\ consumption_{T\ or\ TS} - Energy\ consumption_{TE}}{Energy\ consumption\ _{T\ or\ TS}} \times 100 \tag{24}$$

$$Congestion\ avoidance\ \% = \frac{NP_{TS} - NP_{T\ or\ TE}}{NP_{TS}} \times 100 \tag{25}$$

Notation: NP is the total number of times passed through any congested path.

Table 3. The example problems

No.	Pick list (Assigned vertex no.)	Congested paths	Weight (kg) (Respectively)
1	D12 (15)	0-5, 5-0, 14-9	52
2	E4 (15)	5-0, 1-6, 2-7, 10-15	16
3	G3(15)	11-6	58
4	F11(15)	1-6,7-2,4-9,10-5,13-8,14-9	30
5	B1(15)	0-5,9-4,1-15,15-1	18
6	F8(15)	6-1,12-15,9-14	7
7	D8(15)	1-6, 8-3, 9-15, 15-9, 14-9, 9-14	91
8	B12(15)	6-1, 3-8	65
9	C1(15)	1-6, 7-12, 14-9	98
10	E2(15)	4-9, 9-14	92
11	C10(15), F1(16)	0-5,1-6,7-12, 14-9	50, 78
12	D3(15), G4(16)	0-5,3-15,10-5,11-6, 16-12, 8-13	62, 49
13	F4(15), G3(16)	5-0, 5-10, 8-3, 9-4	98, 15
14	F2(15), C6(16)	16-2, 10-5, 8-13	99, 69
15	D2(15), A4(16)	1-6, 2-7, 7-2, 9-14	85, 33
16	E1(15), G2(16)	1-6, 10-15, 9-14	62, 59
17	A5(15), G4(16)	12-16,16-12	58, 34
18	E1(15), G5(16)	1-6, 6-1, 8-3, 9-4	1, 39
19	F10(15), A4(16)	10-9	46, 99
20	C4(15), D10(16)	16-9, 9-16, 7-12	48, 1
21	D9(15), C9(16), C8(17), A7(18), B5(19)	7-2, 4-15 ,15-4	64, 54, 17, 90, 79
22	G3(15), C2(16), A3(17), E7(18), C12(19)	6-16, 13-8, 8-13	91, 10, 84, 16, 26
23	H11(15), F11(16), A3(17), G9(18), E3(19)	6-11, 7-12, 15-14, 14-15, 9-4	20, 53, 61,18, 71
24	B8(15), C8(16), D11(17), H10(18), A5(19)	0-19, 10-5, 6-1, 15-2, 17-4, 9-18	51, 42, 45, 24, 82
25	C4(15), G9(16), A1(17), H10(18), C2(19)	17-5, 1-6, 19-2, 9-4, 11-6, 8-16, 16-13	18, 64, 67, 20, 75
26	H7(15), B3(16), B6(17), E9(18), A3(19)	5-10, 5-19, 6-17, 7-12, 4-9	85, 53, 66, 93, 85
27	C9(15), C11(16), F3(17), B12(18), D8(19)	0-5, 5-10, 1-6, 6-17, 18-2, 19-14	56, 61, 37, 37, 96
28	H1(15), H11(16), A2(17), G4(18), B2(19)	19-6, 11-6, 2-7, 15-13	69, 24, 37, 97, 61
29	D5(15), E4(16), A10(17), C7(18), C12(19)	18-2, 18-19, 16-5, 7-12, 8-13, 13-8	92, 82, 4, 72, 7
30	A7(15), H4(16), G1(17), D6(18), B11(19)	15-1, 15-6, 19-2, 17-12, 16-13	2, 73, 68, 45, 82
31	B5(19), C12(16), D8(21), D9(18), D10(24), E2(15), F2(22), G1(23), H1(20), H7(17)	0-5, 2-16, 9-4, 15-10, 23-12, 17-14	21, 93, 84, 43, 65, 56, 50, 52, 11, 55
32	C5(20), C9(19), C11(22), C12(24), D9(18), D12(15), F5(21), G4(16), H1(17), H11(23)	1-6, 6-21, 21-6, 16-7, 17-8, 8-3	39, 6, 2, 45, 22, 63, 86, 73, 23, 6
33	A2(16), B4(24), C2(17), C3(15), C4(20), D6(21), D9(18), D12(23), F1(19), H9(22)	24-1, 20-7, 17-2, 21-3, 21-8, 13-8, 9-22	6, 38, 42, 31, 30, 40, 89, 50, 84, 33
34	A2(18), A3(24), B1(19), B2(16), B3(23), D5(17), D10(21), F4(20), G3(15), H1(22)	18-24, 24-5, 6-20, 11-20, 22-13, 17-8	5, 87, 39, 79, 6, 90, 20, 46, 61, 59
35	A6(15), B3(22), B6(23), C1(24), C4(18), D1(20), D5(16), E2(17), F3(21), H1(19)	5-0, 2-24, 16-3, 21-11, 12-7, 8-19, 14-9	49, 82, 17, 20, 57, 13, 61, 94, 56, 81
36	A4(18), B5(23), C2(17), C4(16), D2(22), D12(19), E1(20), F2(15), G4(21), H1(24)	1-23,16-17,8-22,9-19,20-5	34, 31, 37, 59, 6, 19, 62, 86, 68, 72
37	B1(17), B5(15), B6(21), C1(24), D5(20), D6(16), E2(23), F4(18), G3(19), H7(22)	0-5,15-21,6-18	7, 86, 35, 5, 84, 51, 49, 42, 45, 23
38	A1(24), A4(19), A5(22), B6(20), C4(15), E5(18), F5(17), G4(23), H5(16), H11(21)	20-1,15-7,9-4,18-5,7-23,9-21,21-14	58, 13, 83, 2, 48, 17, 9, 61, 13, 28
39	A3(17), A5(23), B2(22), B3(24), C2(19), D2(16), F4(18), F9(21), H5(15), H10(20)	24-6,12-21,21-12,13-15,9-20	83, 47, 20, 81, 94, 48, 67, 40, 87, 32
40	A2(17), B3(15), D2(22), D7(21), D9(18), E1(19), F4(16, F7(23), H5(20), H11(24)	15-6,2-7,22-8,8-22,18-9,6-16,24-14	3, 37, 78, 17, 78, 56, 56, 37, 72, 32, 62

6. Results and discussion

The validation of the results obtained from the proposed use-case models (T and TE) and TS with the results of the brute-force search (BF) are shown in Figs. 10–12. These comparisons illustrate that the solutions obtained by the models are truly optimal and unaffected by any errors in the solver package or programming code, as the results are all consistent with the optimal results of the BF method.

The travel time and the energy consumption of the T, TE, and TS models are shown in Figs. 13–14, and the travel time and energy savings percentages are provided in Tables 4–6. The comparison of the congestion avoidance percentages of the T and TE use-case models with the baseline TS model is shown in Table 7.

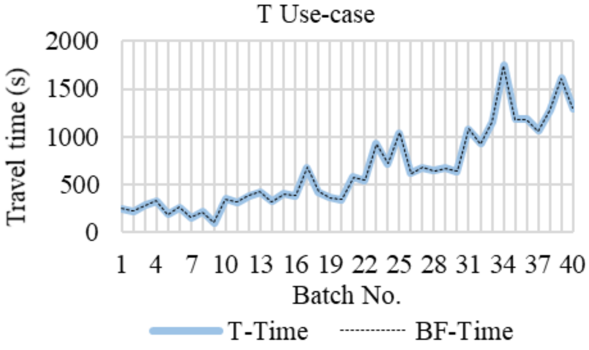


Figure.10 Validation of the optimal T

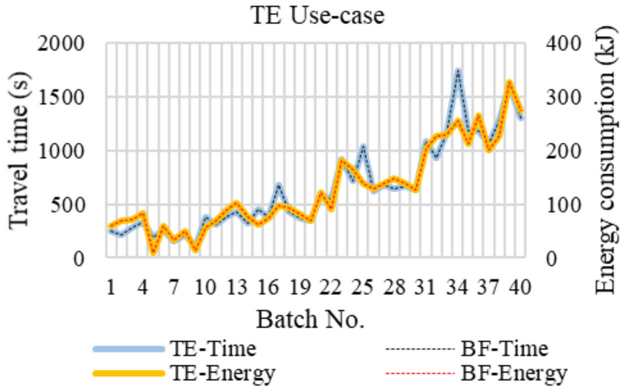


Figure.11 Validation of the optimal TE

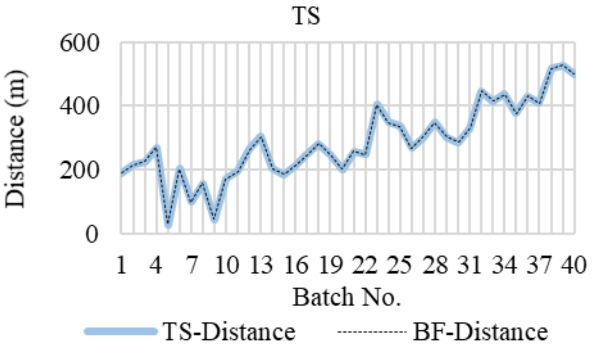


Figure.12 Validation of the optimal TS

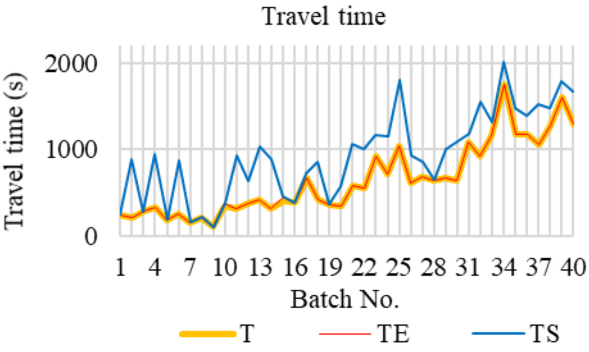


Figure.13 The travel time for T, TE, and TS

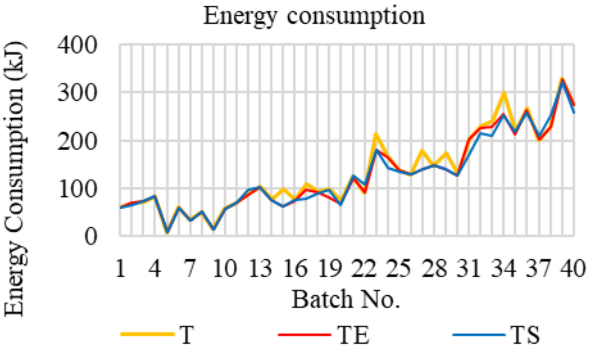


Figure.14 The energy consumption of T, TE, and TS

Table 4. Total travel time savings of T and TE over TS

Batch size	T - travel time savings (%)	TE - travel time savings (%)
1	44.91	44.91
2	40.92	40.18
5	37.51	37.51
10	18.73	18.73
All	30.27	30.04

Table 5. Total energy savings of T and TE over TS

Batch size	T - energy savings (%)	TE - energy savings (%)
1	-0.98	-0.98
2	-8.26	0.05
5	-9.12	-0.02
10	-5.13	-2.24
All	-6.30	-1.20

Table 6. Energy savings of TE over T

Batch size	energy-savings (%)
1	0
2	8.06
5	17.59
10	2.75
All	4.86

Table.7 Congestion avoidance of T and TE compared to the optimal route of TS

	Total congestion passing of TS	Total Avoidance	Avoidance (%)
T	66	43	65
TE	66	38	58

Comparing the travel time of the T, TE, and TS models, it can be observed that the TS model requires a great deal of time, while our proposed model requires less: the maximum travel time savings for the T and TE use-cases are around 44% and 30%, respectively, for all batch sizes. When comparing the energy consumption of the T, TE, and TS models, it can be noted that although the energy consumption of the TS model is lowest, this savings represents a trade-off with the considerable amount of travel time required, in contrast with the TE use-case model, which provides the same level of savings with a shorter travel time.

Comparing the T and TE use-cases, the T use-case provides a slightly lower travel time as the TE use-case trades off a little travel time to save more energy, as shown in Table 6. For a batch size of one item, there are no choices for picking order sequencing, and so the energy savings percentage is zero. The energy savings begin with batch sizes of two or more items. The average energy savings percentage is 17% for a batch size of five items and 4.86% for all picking cases. It can be seen in Table 7 that, compared with the optimal route determined by the TS model (which does not consider the traffic condition), the T use-case avoids 65% of the congested paths and the TE use-case avoids 58%.

The results of the numerical experiment demonstrate that the TE use-case, which represents the full functionality of our proposed model, provides 30% faster picking speed than the TS model and reduces the energy consumption by up to 17% compared with the T use-case. Finally, up to 58% of the congested paths are avoided using the proposed TE use-case model.

7. Conclusion

A significant process in warehouse operation is order picking; it has been identified as the most labor-intensive, time-consuming, and costly activity, and warehousing professionals give order picking the highest priority to improve warehouse productivity. Nowadays, due to either government regulation or market demand, most organizations are beginning to go green. Energy-saving efforts are also identified as one of the most important tasks for green supply chain implementation. Many studies have focused on order picker routing but there exists a lack of research that focusses on simultaneously increasing order picking speed and reducing the energy used in rectangular warehouses with varying levels of traffic in each aisle.

This paper accordingly introduced a new mathematical model for order picker routing that minimizes the total travel time and picking vehicle energy consumption in a rectangular warehouse with varying levels of traffic in each aisle. After formulating the proposed mathematical model, we developed software to simulate the sample problems. The results of these simulations were then validated using the brute-force search method, indicating that the proposed model provides truly optimal routes. We then verified the energy savings by comparing two use-cases: one considering only time (T) and the other considering both travel time and energy consumption (TE). Moreover, the time-staged model (TS) proposed by Letchford, Nasiri, and Theis [13] was utilized as a baseline to verify the travel time savings and congested path avoidance. We found that both use-cases provided routes up to 44% faster than the TS and avoided more than 50% of congested paths. Comparing the two use-cases, the TE use-case consumed less energy due to its preference for picking heavy items at the end of the trip, making it possible to save up to 17% of the energy consumed in the T use-case. The TE use-case, which represents the full functionality of our proposed model, satisfies the objective of simultaneously increasing the order picking speed and reducing energy consumption. From the perspective of efficiency verification, although only a small amount of energy can be saved during a single trip, the implications of these relatively minor savings can be quite large, especially in modern warehouses where thousands of orders are picked every day.

However, only small-scale warehouse has been examined, the large problems are still hard to solve. For future research, we plan to investigate further the capability of the model to perform in a large-scale warehouse and find a suitable calculation technique that conforms with the larger problems.

Acknowledgments

This research was supported by the Faculty of Engineering at Thammasat University. We would like to thank the anonymous reviewers, who have given valuable suggestions and comments that improved the quality of this paper, and our colleagues, who provided insight and expertise that greatly assisted our research.

References

[1] R. De Koster, T. Le-Duc, and K. J. Roodbergen, "Design and control of warehouse order picking: A literature review", *European Journal of Operational Research*, Vol.82, No.2, pp.481-501, 2007.

[2] J. A. Tompkins, J. A. White, Y. A. Bozer, E. H. Frazelle, and J. M. A. Tanchoco, *Facilities Planning*, 4th Edition, John Wiley & Sons, Hoboken, N.J., 2010.

[3] R. Howes, J. Skea, and B. Whelan, *Clean and Competitive: Motivating Environmental Performance in Industry*, 1st Edition, Routledge, London, 2013.

[4] A. Hafezalkotob, "Competition, cooperation, and coopetition of green supply chains under regulations on energy saving levels", *Transportation Research Part E: Logistics and Transportation Review*, Vol.97, pp.228-250, 2017.

[5] T. Minav, L. Laurila, and J. Pyrhönen, "Energy recovery efficiency comparison in an electro-hydraulic forklift and in a diesel hybrid heavy forwarder", In: *Proc. of International Symposium on Power Electronics, Electrical Drives, Automation and Motion*, pp.574-579, 2010.

[6] F. Chen, G. Xu, and Y. Wei, "Heuristic routing methods in multiple-block warehouses with ultra-narrow aisles and access restriction", *International Journal of Production Research*, Vol.57, No.1, pp.228-249, 2019.

[7] U. Pferschy and J. Schauer, "Order batching and routing in a non-standard warehouse", *Electronic Notes in Discrete Mathematics*, Vol.69, pp.125-132, 2018.

[8] S. Quader and K. K. Castillo-Villar, "Design of an enhanced multi-aisle order-picking system considering storage assignments and routing heuristics", *Robotics and Computer-Integrated Manufacturing*, Vol.50, pp.13-29, 2018.

[9] R. De Koster and E. Van Der Poort, "Routing order pickers in a warehouse: a comparison between optimal and heuristic solutions", *IIE Transactions*, Vol.30, No.5, pp.469–480, 1998.

[10] J. A. Cano, A. A. Correa-Espinal, and R. A. Gómez-Montoya, "An evaluation of picking routing policies to improve warehouse efficiency", *International Journal of Industrial Engineering and Management*, Vol.8, No.4, pp.229-238, 2017.

[11] K. J. Roodbergen and R. De Koster, "Routing order pickers in a warehouse with a middle aisle", *European Journal of Operational Research*, Vol.133, No.1, pp.32–43, 2001.

[12] L. Pansart, N. Catusse, and H. Cambazard, "Exact algorithms for the order picking problem", *Computers & Operations Research*, Vol.100, pp.117-127, 2018.

[13] A. N. Letchford, S. D. Nasiri, and D. O. Theis, "Compact formulations of the Steiner traveling salesman problem and related problems", *European Journal of Operational Research*, Vol.228, No.1, pp.83–92, 2013.

[14] J. Y. Shiau and T. C. Liao, "Developing an order picking policy for economical packing", In: *Proc. of IEEE International Conference on Service Operations, Logistics, and Informatics*, pp.387–392, 2013.

[15] D. G. Ramirez-Rios, L. P. M. Romero, and J. R. Montoya-Torres, "Multicriteria optimization in a typical multi-aisle warehouse with multiple racks", In: *Proc. of EURO Mini-Conference on Optimization in the Natural Sciences*, pp.35–48, 2014.

[16] M. Çelik and H. Süral, "Order picking in a parallel-aisle warehouse with turn penalties", *International Journal of Production Research*, Vol.54, No.14, pp.4340–4355, 2016.

[17] T. Chabot, R. Lahyani, L. C. Coelho, and J. Renaud, "Order picking problems underweight, fragility and category constraints", *International Journal of Production Research*, Vol.55, No.21, pp.6361-6379, 2017.

[18] M. M. Abid and I. Muhammad. "Heuristic approaches to solve traveling salesman problem", *Indonesian Journal of Electrical Engineering and Computer Science*, Vol.15, No.2, pp.390-396, 2015.

[19] R. Jadczak, "Traveling salesman problem: approach to optimality", *Przedsiebiorczosc i Zarzadzanie*, Vol.15, No.2, pp.157-169, 2014.

[20] F. H. Prabowo, K. M. Lhaksmana, and Z. A. Baizal, "A Multi-Level Genetic Algorithm Approach for Generating Efficient Travel Plans", In: *Proc. of the 6th International Conference on Information and Communication Technology*, pp.86-91, 2018.

[21] E. Baidoo and S. O. Oppong, "Solving the tsp using traditional computing approach", *International Journal of Computer Applications*, Vol.152, No.8, pp.13-19, 2016.